

**Universitatea Politehnica București
Facultatea de Automatică și Calculatoare**

Proiect Achiziții de Date
Osciloscop in LabView

**Loghin Dumitrel
Neagoe George Dănuț
Simion Ciprian**
(grupa 324CB)

2008

Osciloscop digital in Labview

Introducere

Proiectul a fost realizat în Labview versiunea 8.0 și este compatibil cu dispozitivele de achiziții de date ieftine National Instruments USB(în special NI USB 6009). Interfața grafică este realizată după modelul unui osciloscop clasic. Pentru a simula funcționalitatea unui osciloscop, aplicația preia intrările analogice prin două canale și apoi caută un nivel de trigerare specificate de utilizator. Acest lucru permite formelor de undă periodice să fie aliniate și afișate ca unde de sine stătătoare. Există de asemenea controlere care sunt folosite pentru scalarea semnalelor , reglarea timpului / div și măsurarea offseturilor similar cu un osciloscop real.

Instalare

Pentru a rula programul trebuie să urmați următorii pași:

1. Trebuie să aveți instalat LabVIEW Run-Time Engine versiunea 8.0
2. De asemenea și driver-ul DAQmx driver versiunea 8.0 instalat cu setările implicite (amândouă pot fi descărcate de pe ni.com)
3. Conectați dispozitivul NI USB 6009 la calculator(sistemul de operare ar trebui să detecteze și să configureze dispozitivul – se poate să apară o fereastră care să vă zică să instalați USB-6XXX Firmware Loader) și urmați instrucțiunile de pe ecran(dacă este cazul) până când dispozitivul este instalat în calculator.
4. Pentru a rula aplicația trebuie să determinați id-ul dispozitivului de achiziție de date USB-6009 , prin care dispozitivul este identificat de către osciloscop. Acesta îl puteți găsi deschizând programul National Instruments Measurement and Automation Explorer (MAX). La secțiunea Devices and Interfaces => NI-DAQmx Devices. Dacă driverul DAQmx este instalat bine, și dispozitivul USB-6009 este conectat, se va observa instalat “Dev1” unde Dev1 indică dispozitivul 1. Notați id-ul acelui dispozitiv.
5. Când rulați aplicația , va trebui să selectați un număr de identificare(id) corect. Dacă se selectează unul greșit , veți întâmpina un mesaj de eroare. Pentru a corecta acest lucru , asigurați-vă că dispozitivul USB-6009 este conectat, selectați id-ul corect, și apăsați butonul RESET

Conexiunile Hardware

Conectați semnalele de intrare la canalele diferențiale 0 și 1 ale dispozitivul de AD. Terminalele 2 și 3 corespund intrărilor pozitive și negative pentru canalul diferențial 0. Terminalele 5 și 6 corespund intrărilor pozitive și negative pentru canalul diferențial 1. Pentru mai multe informații asupra terminalelor și conexiunilor , consultați manualul dispozitivului.

Controale

Dispozitiv: Specifica Numarul de Identificare al USB 6009, care este unic. Acest număr poate fi găsit în Measurement and Automation Explorer (MAX). Dacă reglați această intrare , va trebui să apăsați butonul RESET pentru ca schimbările să se producă.

Canale: Specifică numărul de canale care se vor scana. Prin selectarea unui singur canal de date, se poate obține o rată dublă comparată cu rata obținută prin selectarea a două canale. De asemenea trebuie apăsat butonul RESET pentru ca schimbările să intre în vigoare.

Rată de Voltaj Intrare: Setează amplificarea internă a amplificatorului programabil a dispozitivului 6009. Aceasta aplică o amplificare potrivită semnalului care se pretează digitizării. Pentru rezultate

mai bune folosiți cea mai mică rată care practic încapsulează semnalul așteptat. Trebuie să apăsați de asemenea RESET pentru actualize setările făcute.

Reset: Dacă una dintre valorile controalelor de mai sus este modificată, aplicația trebuie să fie resetată pentru ca schimbările să aibă effect.

Aprinde/Stinge: Schimbă vizibilitatea graficului fiecărui canal(nu afectează procedeul de citire de la dispozitivul hardware ci doar permite utilizatorului să efectueze operația de afișare/ascunderea a datelor)

Abatere: Reglează poziția vertical a curbei pe ecran adaugând sau scăzând o abatere al voltajului.

Zero: Șterge orice abatere adăugată cu ajutorul controlului Abatere.

Centrare: Setează automat abaterea astfel încât forma de undă să fie centrată vertical pe afișor.

Scală : Setează numărul de Volți / diviziune pe axa verticală pentru a realize un “zoom” vertical

Timp: Setează timpul / diviziune pe axa orizontală pentru a face “zoom” orizontal. Aceasta nu schimbă rata de achiziție a datelor de la hardware.

Canal de Trigerare: Specifică ce canal va genera trigger-ul

Front: Schimba frontul pe care se realizeaza trigerarea (crescator / descrescator).

Nivel: Specifică nivelul de voltaj la care va fi generat trigger-ul. Dacă nivelul nu este intersectat de semnalul periodic , trigger-ul nu va funcționa iar forma de undă nu va fi afișată correct. De asemenea trigger-ul este definit in software, și nu va afișa corect decât semnalele periodice cu o perioadă mai mică decât 1/8 secunde.

Cursoare (On/Off): Afișează / Nu cursoarele.

Blochează Cursoare: Determină dacă cursoarele sunt libere sau atașate unui canal. Dacă acestea sunt libere atunci voltajul nu va fi afișat în dreptul lor.

C1: Dacă opțiunea Blochează Cursoare este activă , selectează canalul de care este atașat Cursorul 1.

C2: La fel ca la C1 numai pentru Cursorul 2

Detalii Realizare

Realizarea acestui proiect a pornit de la exemplele NI (Help). Pentru adaptarea la microcontrolerul USB 6009 am utilizat documentație suplimentară de pe Internet (de la National Instruments). Aplicația este realizată modular, acest lucru ajutând la depanare, dar și la o implementare specifică lucrului în echipă.

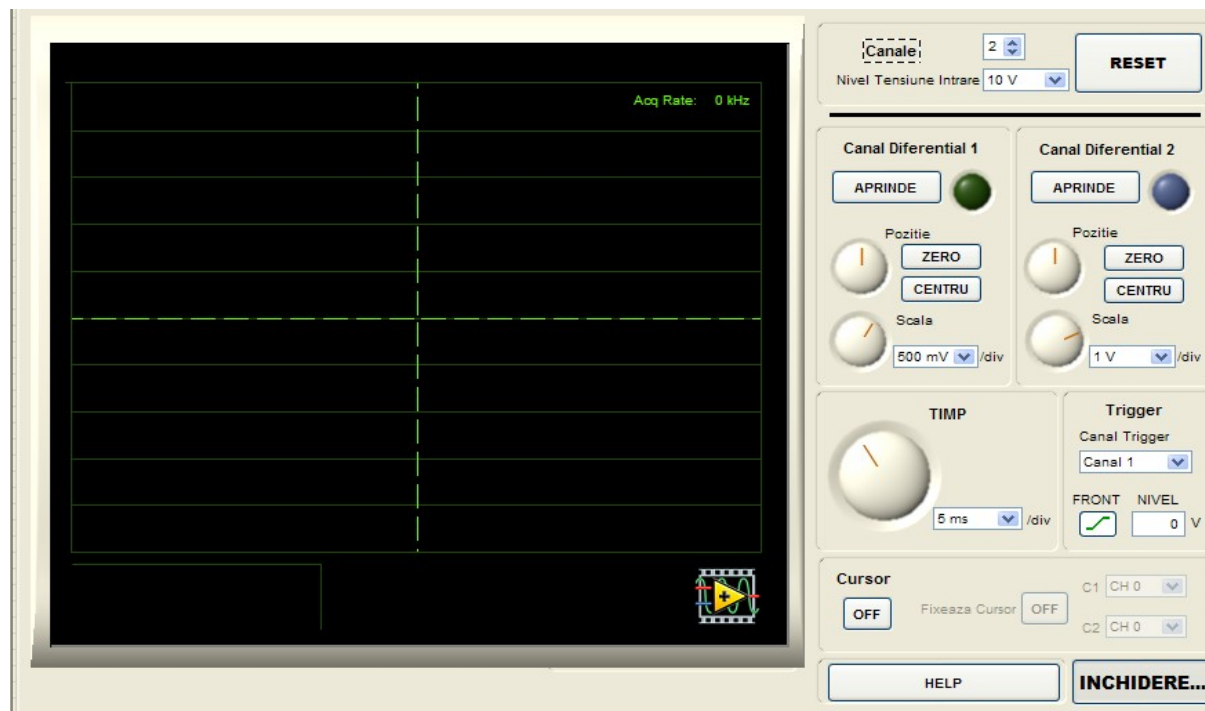


Figura 1. Aspect aplicație

Modulele aplicației sunt:

- Osciloscop.vi – modulul principal, în care sunt utilizate și celelalte module
- Loading.vi – modulul de încărcare
- Time base convert.vi – modul conversie timp
- Voltage scaling.vi – modul scalare tensiune
- Voltage range convert.vi – modul conversie nivel tensiune
- Voltage convert.vi – modul conversie tensiune
- Trigger.vi – modul trigger
- Two ch update.vi – modul actualizare canale
- Format cursor values.vi – modul formatare date obținute din poziția cursorilor
- Calculate cursor values.vi – modul determinare date din pozițiile cursorilor
- Help.vi – modul reactualizabil de documentație

Vom descrie mai întâi fiecare modul, iar la final vom descrie funcționalitatea modulului principal.

1. Loading.vi

Afișează o bară de încărcare până se inițializează driver-ul și se fac calculele preliminare de desenare a graficului. Acest lucru este realizat în cadrul unui ciclu. De asemenea utilizăm și un **timer** de 50 ms (timp suficient pentru inițializare). Afișajul încărcării se realizează utilizând o funcție numerică numită **Quotient & Remainder**.

2. Time base convert.vi

Realizează conversia din **secunde** în **milisecunde** (utilizate la reglarea **time/div** din cadrul osciloscopului). Este utilizat un șir indexat (**Index Array**) ce conține valorile pentru timp. Cum selecția vine de la un control cu ieșire întreagă, modulul realizează și conversia în real.

3. Voltage scaling.vi

Modulul care realizeaza reglarea **V/div** și a poziției (**Y**) în cadrul osciloscopului. Acest lucru este realizat pentru ambele canale independent. Algoritmul este următorul(pentru un singur canal):

Dacă este selectat ZERO (de pe panoul frontal)

Se setează 0 la cursorul de poziție (**Knob**) (pentru acesta se utilizează o referință)

Se trimitte valoare la șirul de ieșire

Altfel

Dacă este selectat CENTRU

Este preluat subsirul pozițiilor corespunzătoare scalei curente

Este făcută o medie a elementelor (apoi este trecută cu semn schimbat)

Se setează cursorul la valoare calculată

Se trimite la ieșire valoarea

Altfel

Este preluată direct și trimisă valoarea cursorului de poziție

Sfârșit

Iată și programul LabView (corespunzător primei ramuri Dacă):

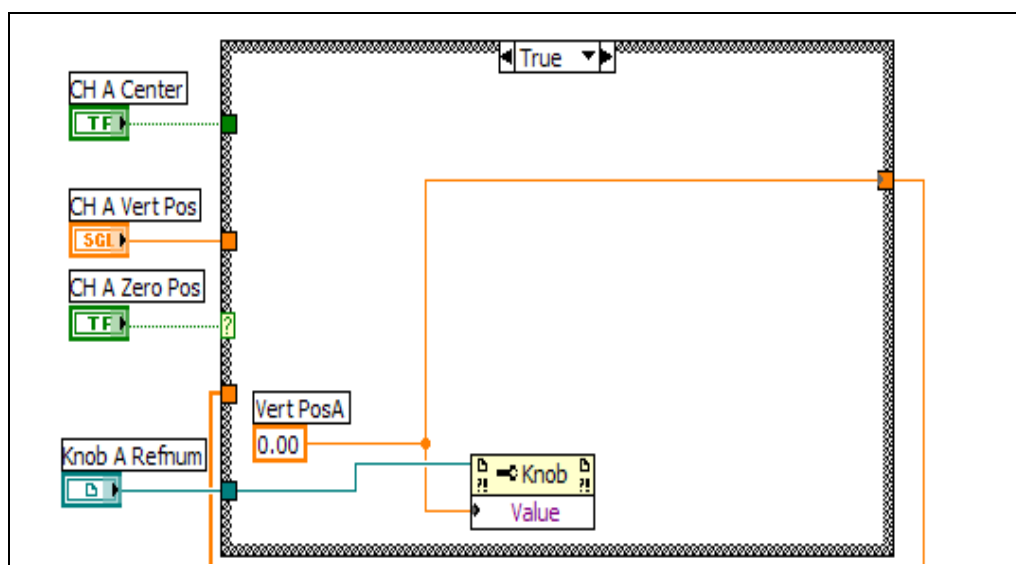


Figura 2. Modul scalare tensiune

Acest modul folosește modulul de conversie tensiune pe care îl vom descrie mai jos. Acesta este utilizat pentru a selecta corespunzător valorile poziției verticale.

4. Voltage convert.vi

Acest modul seamănă cu cel de timp. Realizează conversia în V a tensiunii utilizând un Index Array. Valoarea la ieșire e reală.

5. Voltage Range Convert.vi

Realizează setarea intervalului de tensiuni suportate la intrare și calculează Vmin și Vmax care vor fi folosite de osciloscop. Utilizăm de asemenea un **Index Array**. Valorile suportate sunt:

- 1 V
- 5 V
- 10 V
- 20 V

6. Trigger.vi

Elementul central este **Basic Level Trigger Detection**. Acesta acceptă la intrare nivelul, frontul, canalul, toate preluate de pe panoul osciloscopului. La ieșire furnizează dacă o valoare booleană care precizează dacă a fost descoperita condiția de declansare, precum și zona de apariție a declanșării.

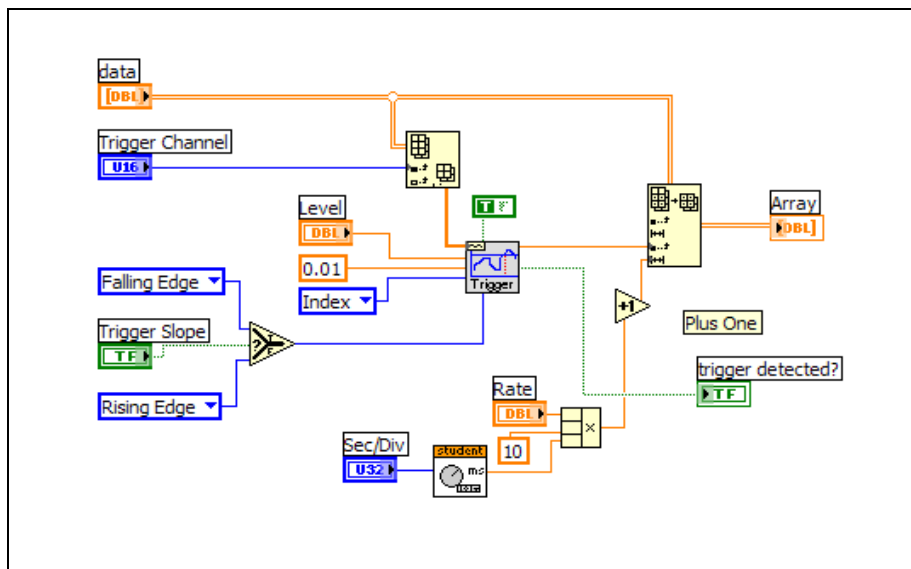


Figura 3. Trigger

7. Two ch update.vi

Acest modul realizează practic resetarea axei X și a cursorilor când apare un eveniment de schimbare a unei valori. Astfel, dacă este schimbată scalarea timpului, se resetează axa X în conformitate cu noua valoare și, de asemenea cursorurile sunt readuse la centru.

8. Calculate cursor values.vi

În acest modul se calculează valorile:

- C1 Voltage – valoarea tensiunii indicate de cursorul 1

- C2 Voltage – valoarea tensiunii indicate de cursorul 2
- dV – diferența de tensiune
- dT – diferența de timp

Principiul este simplu: din referința graficului se preiau pozițiile celor două cursoare. Cu ajutorul funcțiilor diferență, modul, împărțire se calculează dT, dV, C1 și C2. Sunt folosite și niște șiruri (Index Array) care furnizează valorile corespunzătoare anumitor stări.

Ieșirile sunt furnizate modulului de formatare descris mai jos.

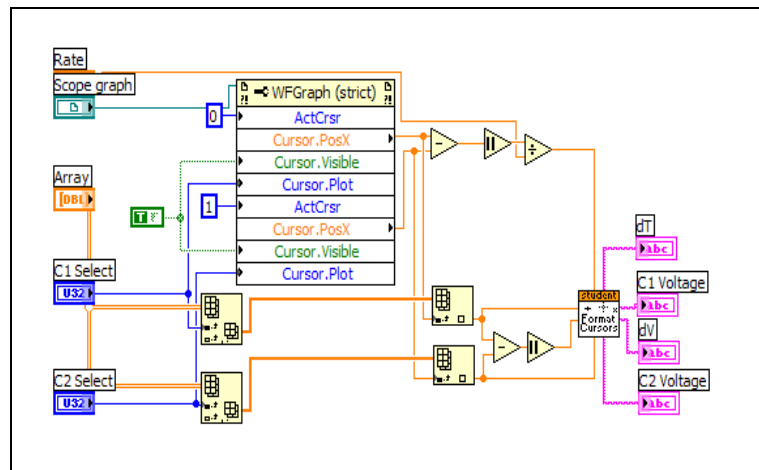


Figura 4. Calcul cursorare

9. Format cursor values.vi

Acest modul realizează formatarea datelor sub o formă ușor de înțeles și în același timp științifică. Este utilizată o structura **Number To Engineering String** care formatează datele sub formă științifică (exprimare cu E). Apoi un **Match Pattern** adaugă unitățile de măsură în funcție de mărime. Ieșirile sunt sub formă de **String** și vor fi afișate pe grafic.

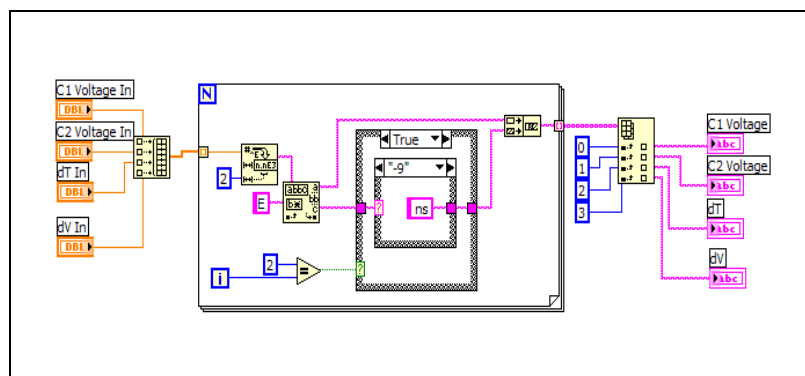


Figura 5. Formatare date

10. Help

Modulul conține un element de afișare cu scroll bar. Conținutul este păstrat de un **String**. Acest modul este afișat la apăsarea butonului de **HELP** și conține un mic ghid de utilizare a osciloscopului.

Osciloscop.vi (modulul principal al aplicatiei)

Vom descrie acest modul pe blocuri, după cum acestea apar în programul LabView.

1. Inițializare

Este realizat într-o structură secvențială (**Stacked Sequence Structure**).

- Cadrul 0
Este legat la modulul de încărcare. Proprietățile sunt setate prin referințe.
- Cadrul 1
Realizează inițializarea osciloscopului prin setarea la valori predefinite. Printr-un element de tipul **Build Array** trimite setările modulelor vizate.
- Cadrul 2
Inițializează afișaje, canale, cursoare la valori implicite.
- Cadrul 3
Initializeaza zoom, scrolling.

2. Ciclu principal

Aplicația se oprește în situațiile:

- Se apasă butonul **IEȘIRE**
- Se apasă butonul **X** al ferestrei (ieșire forțată).
- S-a produs o eroare internă (vezi firele roz care reprezintă erorile).

3. Blocul de selecție canale

Osciloscopul poate funcționa cu unu sau două canale. Blocul de selecție este de asemenea o structură secvențială.

La intrare prezintă un **DAQmx Create Virtual Channel**. Este creat astfel un model virtual de instrument de achiziție de date. Sunt setate:

- Mod de lucru (diferențial / nediferențial)
- Nivelul de tensiune (apelează la modulul **Volts to Range Convert**)
- Erorile sunt transmise mai departe (firele roz)
- Task-urile (sarcinile) sunt transmise prin firele mov.

Numarul de canale se setează prin controlul prezent pe panoul osciloscopului. Structura primește acest număr și prezintă cadrele (practic este un switch):

- 1 (canal)
Este valoarea implicită. În acest cadru se setează prin intermediul unei referințe (**Property Node**) numărul de afișaje. Canalul 2 este deselectat (disabled). De asemenea în această diagramă apar setările pentru achiziție.
- 2 (canale)
Sunt setate ambele canale și, de asemenea, ambele zone de afișaj grafic. Sunt setate și în acest caz ratele de achiziție pentru 2 canale.

La ieșire avem un **DAQmx Timing** care realizează un buffer cu eșantioanele preluate, dacă este cazul. Este comandat de task-urile **DAQmx Create Virtual Channel**.

4. Ciclu prelucrare semnal

Este comandat pe intrare de un **DAQmx Control Task** care preia task-urile de la elementele anterioare. În interior există:

- **DAQmx Start Task** – dă semnalul de începere a măsurării și prelucrării
- **DAQmx Read** – citește eșantioanele și furnizează o ieșire numerică reală. Practic este un convertor. Datele citite sunt plasate într-o coadă
- **DAQmx Stop Task** – opreste achiziția
- **DAQmx Clear Task** – șterge sarcinile și emite erori, dacă acestea au aparut.

Ciclul se opreste când:

- Apare o eroare (status)
- Se apasă **RESET**
- Se apasă închidere **STOP**

Tratarea erorilor

După cum am mai precizat, erorile sunt unificate și transmise spre un afișor. Se utilizează structuri **Merge Errors** care unifică erorile apărute în diverse surse.

5. Coadă de semnale

Sunt utilizate elementele:

- **Obtain Queue** – crează o referință la un obiect de tip coadă.
- **Enqueue Element** – adaugă un element la coadă. Dacă aceasta este plină așteaptă un timp (timeout).
- **Get Queue Status** – verifică starea curentă a cozii.
- **Dequeue Element** – preia un element din coadă.

6. Ciclu control și afișare

Intrarea este reprezentată de datele din coada de semnale. Utilizează algoritmul:

Verifica stare coadă

Dacă sunt mai mult de 2 elemente

Golire coadă

Întoarce șir valori

Preia primul din coda (cu un timeout de 650 ms)

Apelare modul Trigger, cu intrările (preluate din controalele de pe panou)

- Nivel
- Canal
- Timp

Ieșire sunt preluate de modulul de Voltage Scaling și de cadrul de control cursoare.

7. Cadrul cu cursoare

Setat de butoanele **ON / OFF**. Cursoarele sunt folosite la determinarea valorilor în diferite puncte ale graficului.

Dacă este setat **ON (true)**

Se apelează **Calculate Cursor Values**.

Altfel

Se ascund cursoarele (printr-o referință)

8. Grafic

Aceasta parte utilizează modulele **Time base convert** și **Two Ch update** descrise mai sus.

9. Structura control

Contine 8 cadre

- Cadru 0
Corespunde apelului modului **Help**.
- Cadru 1
Sesizează schimbările și recalculează valorile cursorilor.
- Cadru 2
Gestionare timeout.
- Cadru 3
Gestionare evenimente RESET si STOP.
- Cadru 4
Gestionează închiderea normală a aplicației, eliberând resursele folosite.
- Cadru 5
Fixare cursori.
- Cadru 6
Gestionează desenarea pe canalul 1
Seteaza Active Plot.
Seteaza led-ul verde.
- Cadru 7
Gestioneaza desenarea pe canalul 2
Seteaza led-ul albastru.

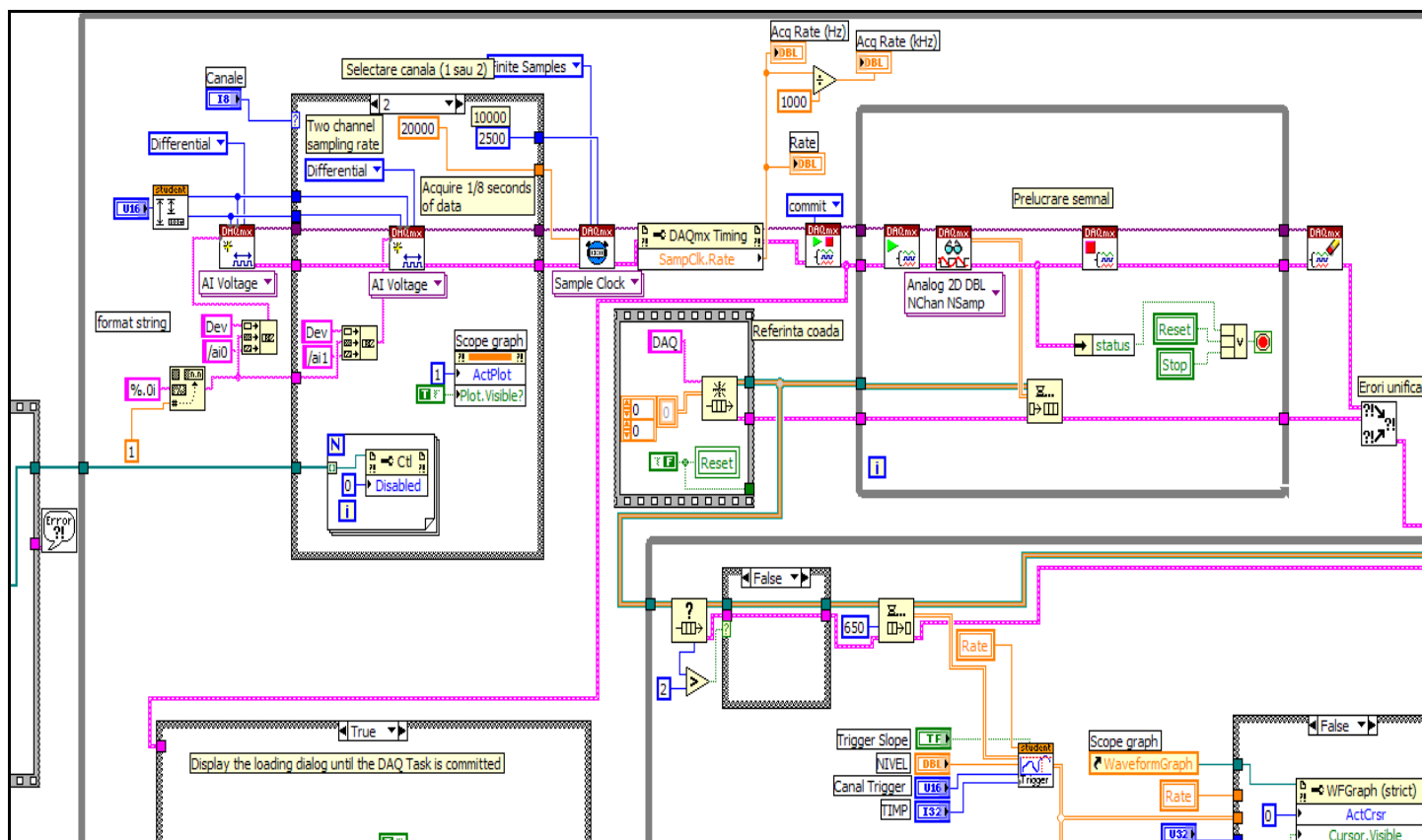


Figura 6. Parte din programul LabView principal